COMBATING ROTATION, DILATION AND TRANSLATION IN DIGITAL WATERMARKING

By

Yeung Siu Wai

A Thesis

submitted in partial fulfillment of the requirements

for the Degree of Master of Philosophy

Division of Information Engineering

The Chinese University of Hong Kong

July 2000

Acknowledgement

I would like to take this opportunity to thank my supervisor Prof. Victor Keh-wei Wei for his guidance, introducing me to the image processing as well as cryptography, and supervising my graduate study.

Furthermore, I would like to express my gratitude to all staff, all postgraduate students, and all undergraduate students of the Department of Information Engineering who gave me much help and the most unforgettable days in school life during these two years.

In addition, I wish to express my appreciation to Prof. Lian-kuan Chen, Prof. Wing-shing Wong and Prof. Michael Ming-yuen Chang who enriched my knowledge in signal and system and modern communication when I was a tutor of their courses.

Special thanks to all of my friends who gave me much spiritual suppoer and encouragement whenever I was sad and frustrated in doing research.

Last but not least, I greatly thank my parents and my little sister for their full support, patience and many other things to write down.

摘要

電子媒體發展一日千里，其完美複製及網上傳送的優點卻引申非法複印的問題。電子水印爲其中一個應付非法複印的方法，雖然電子水印並不能直接防止非法複印，不過卻可以成爲指証非法複印的証據。

一個設計完善的電子水印需要在各種攻擊下仍能生存。本論文的焦點就集中在相片電子水印的幾何攻擊(geometric distortion)方面，這些幾何攻擊包括旋轉(rotation)，擴大縮小(dilation)及相片微粒推移(image pixel translation)。面對這些攻擊主要有兩個方法，其一爲設計獨特的電子水印。獨特的電子水印包括環形水印(ring-shaped watermark)[24]，phase Taylor invariance 水印以[25]及幾何攻擊免役水印[23]，它們都是一些特別設計可以在幾何攻擊之下仍然生存的水印。另一種面對幾何攻擊的方法就是設法偵測攻擊的幅度，然後將其糾正，當中可以應用相位角對比(phase angle comparison)之技巧令到偵測更準確。此方法的優點在於無需重新設計新的水印，所以一些現有設計良好的水印可以繼續使用。

若各種的幾何攻擊混合出現，偵測的過程將會變得更加複雜。其偵測之方法將是需要繼續研究之項目。

Abstract

With the growth of digital media in the past years, the advantages of perfect duplication and convenient transmission of digital media give a problem of bootlegging of digitized data. Digital watermarking is one of the possible ways to cope with bootlegging. Although it cannot prevent piracy directly, it can be evidence of piracy when they are caught.

There are distortions that a well-designed watermark should be robust to. This thesis is focus on combating the one of these distortions: geometric distortions on digital image-watermarking. The geometric distortions discussed here include rotation, dilation (image rescaling) and translation (image pixel shifting). There are two techniques discussed in this thesis to combat geometric distortions, namely, special designed watermarking systems and distortion detection. Special designed watermarking systems include ring-shaped watermark [24], watermarking with phase Taylor invariance [25] and rotation, dilation and translation invariant watermark [23]. They are specially designed watermarks which are resistant to geometric distortions. The second technique, distortion detection, detects the magnitude of geometric distortion and corrects the distortion by rotate, dilate and translate the image back. The advantage of this technique is that no new watermarking algorithm is needed to be designed.

The existing watermarking algorithms that proven to be robust against other distortions can still be used, with an additional robustness against geometric distortions can be gained.

For the combined geometric distortion, the process of distortion detection is much complicated, especially for combination of translation and rotation distortions. A combined translation and rotation attack is equivalent to a center-unknown rotation and cannot be detected efficiently. This kind of mixed attack is left as further work.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Bootlegging is always a problem for the music, image or video creators. Traditionally, bootleggers' customers have had to put up with second rate products. However, with the development of digital media in the past years, bootlegging had become a convenient and simple task. Digital media has advantages over analog media: the quality of digital data is higher than that of their counterpart. Moreover, digital data can be duplicated without the loss of fidelity. As a result, once a music, image or video product is digitized, they can be pirated perfectly and the pirated product can be identical to the original copy. Digital media also has its advantages on the transmission and access of data over network: they are much easier and simpler in transmission and access. Therefore, the digitized pirate product can be distributed through Internet in mass-scale.

As creators of these digitized products, they have to protect their ownership. Conventional cryptographic systems can help them prevent any unauthorized users from accessing the encrypted data. However, once the data is

decrypted, unauthorized duplication and transmission cannot be prevented anymore. As a result, digital watermark is developed. Digital watermarking is a data hiding technique, which evolved from the ancient technique of *steganography* [1]. *Steganography* means 'covered writing', which is derived from the Greek word *steganos* (means 'covered') and *graphein* (means 'to write'). According to Herodotus, the first Greek historian, and his great work, *The Histories*, idea of covered writing appeared in fifth century BC. In the war between Persia empire and smaller Greece, the art of secret writing saved Greece from being conquered by Xerxes, the king of Persia.

At around 490 BD, Xerxes decided to invade Greece and began to mobilise a military force. He spent five years to build up this force secretly. Finally, in 480 BC, Xerxes prepared to attack Greece in a surprise. However, during this five years, a Greek called Demaratus, who had been expelled from his homeland and lived in a Persian city, witnessed the build-up of the Persian military. Although he had been expelled, he still felt some loyalty to Greece. He wanted to send a message to his homeland to warn the Greek of Xerxes' invasion plan. But he had to face the challenge of transmitting his message secretly without being intercepted by the guards of Persia. He scrapped the wax off a pair of wooden folding tablets, wrote his message on the wood underneath and covered his message by wax again. The tablet appeared to be blink on the surface and thus was not suspected by the Persia guards.

When the tablet reached its destination, no one was able to know the secret hidden. Until a girl called Gorgo was divined and knew that secret was hidden in the tablet. She told the others to scrapped the wax off and the message was revealed. This message was passed all over the Greece and as the results, the

Greeks began to arm themselves. At 480 BC, Xerxes started his inversion. But the Greeks were all well prepared. Finally, Xerxes withdrew his forces at 479 BC.

The story of Persia war demonstrates the use of *steganography*. In nowadays computer world, the idea of *steganography* is used in digital watermark. Digital watermark has the same nature as the paper watermark. It contains hidden information and was imprinted on the product. Once this information is needed, it is revealed. For digital watermark, the information hidden is, at most of the time, the proof of ownership of the product. Because copying and distributing are convenient for digital data on the Internet, mass-scale piracy is easy to happen. Digital watermark can help to protect the ownership of the digital data and prevent the data being bootlegged. Although digital watermark cannot prevent the data from being duplicated directly, it can act as an evidence of bootlegging when the pirate is caught.

Because digital watermark is a proof of ownership of a digital data, if it is removed and disappeared from the data, the ownership proof is lost. As a result, for a well-designed digital watermark, it should be robust to various attacks that an enemy is possible to do on a digital data. In the next chapter, several possible attacks on digital watermark will be mentioned. The main focus of this thesis is on defending one of the attacks in image-watermarking: geometric distortions. The possible defenses on geometric distortion will be discussed in chapter 4 and 5.

Chapter 2

Background

Digital watermark is a perceptually and statistically undetectable secondary signal hiding in the original signal. This secondary signal contains the information of the ownership of the original signal. The owner of the watermark can detect the existence of watermark easily because he knows the exact location of the information hidden. But for bootleggers, they do not know the location of watermark. They cannot extract or delete the watermark out from the data. The only thing they can do is destroying the watermark by adding distortions on the data. As a result, a good watermark should be robust to attacks.

In this chapter, some desirable properties of digital watermark will be listed. It is followed by the description of two possible implementations of digital watermarking system: embedding watermark on spatial domain or frequency domain. Lastly, the capacity issue of digital watermark will be talked in this chapter.

In chapter three, a basic image-watermarking embedding and detecting system will be talked. This system is implemented with the aim of doing tests. It is simple and the algorithm is similar to [12].

In chapter four, the general techniques that can deal with the geometric distortions will be discussed. They are some special designed watermarks that are resistant to geometric distortions. The main focus of this thesis is on specific technique that can cope with geometric distortions, which is distortion detection and correction. A practical way to do distortion detection is phase angle comparison, which will be described in chapter five.

2.1 Properties of a digital watermarking system

Digital watermark began around 1993 with the exploration of simple least-significant bit (LSB) hiding schemes [2]. Afterwards, a large number of digital watermarking schemes are proposed [3][4][5][6][7][8][9][10][11][12][13][14][15][16]. All of them try to achieve the desirable properties of watermark listed below.

1. **Transparency**

   Digital watermark is a secondary signal hiding in the original signal. If the original signal is affected when this secondary signal is embedded, it is undesirable. Therefore, transparency is an important property of this secondary signal. It means that the secondary sign should be perceptually undetectable and should not introduce any interference to the original signal.

2. **Security**

   To make a digital watermark secure, it has to prevent unauthorized users from detecting or altering the embedded watermark. The watermarking

system should be secure and unbreakable unless the unauthorized user knows the secret key that controls the insertion of the watermark. Even if an unauthorized user knows the algorithm of watermark embedding very well, he still cannot break the system. In other words, the security is relied on the secret key of watermark insertion instead of the algorithm's secrecy.

3. **Robustness**

   Even if an unauthorized user cannot detect the presence of watermark, he can destroy the watermark by adding noise or distortions to the digital media. A robust watermark should be difficult to remove. Or any attempts to remove or destroy a watermark result in severe degradation in data fidelity before the watermark is lost.

   There are kinds of attack or distortion that an unauthorized user could add on the digital media that contains watermark. A well-designed watermark should be robust to these attacks:

   (a) *Common signal processing*

   The processes include digital-to-analog or analog-to-digital conversion, resampling, requantization of sample values, compression and decompression, noise addition, filtering, and etc. The watermark should still be detectable and retrievable after these operations are applied.

   (b) *Collision and forgery*

   Collision and forgery are other two kinds of attack on watermark. If an unauthorized user can obtain multiple copies of marked data

and unmarked data simultaneously, he can do collision and forgery attacks on the watermark system. Collision attack is referring the process of averaging multiple copies of data. The data collided can be watermarked and unmarked data, or they can all be watermarked data but have different watermark. All this collisions are possible to weaken the inserted watermark. For forgery attack, the attackers try to forge the watermark that inserted to data. By comparing the differences between watermarked data and unmarked data, the information of watermark can be obtained.

(c) *Geometric distortion*

For image and video data, there are forms of geometric distortion that watermark should be robust to. These distortions include cropping, translation (pixel shifting), dilation (image size scaling) and rotation. These distortions can be applied to the data separately or simultaneously. The main focus of this thesis is on defending these attacks at image-watermark. In the next chapter, the operation details of translation, dilation and rotation will be discussed.

2.2 Digital watermarking in still images: Spatial-domain vs Frequency-domain

For the techniques of embedding watermark in a still image, they can be broadly classified into two categories: digital watermarking based on spatial-domain and frequency-domain. In the earlier watermarking schemes proposed, they were

mostly spatial-domain based watermarking schemes. For instance, in [14], it discussed how to embed a watermark into an image by modifying the less significant bits (LSB) of the image pixels. Modifying LSB of the image pixels was a simple technique in the early research of digital watermarking. There were other improved and varied forms of spatial-domain based watermarking system based on pixel modification [17][15][16]. As shown in these papers, the spatial-domain based watermarks are robust to common signal processing attacks. However, it has disadvantage of low capacity. Capacity is another issue about digital watermarking which will be discussed in section 2.3.

Compare with spatial-domain based watermarking systems, frequency-domain based systems have larger capacity. The proposed schemes include several kinds of common image transforms such as discrete cosine transform (DCT), wavelet transform and Fourier transform [9][7][18][6][5][3][19]. The concept of frequency domain based watermark is embedding the digital watermark by modifying the frequency coefficients after transform instead of modifying the LSB of image pixels. With the help of spread spectrum ideas, the digital watermark can be even more robust [12][20]. In this research, a frequency-domain based watermarking system is implemented in order to perform the tests in this thesis.

2.3 Capacity in digital watermarking

Capacity is concerning the amount of data that can be embedded into an image. The size of the watermark embedded and the number of different watermark can be embedded into an image are directly related to the capacity issue. There are papers that discussed this issue [21][22]. They calculated the theoretical limit on

the capacity and then designed watermarking schemes that approach this limit.

For the spatial-domain based watermarking systems, their capacity is relatively smaller than frequency-domain based systems. It is because the spatial-domain based systems embed the watermark by modifying the LSB of the image pixels. Compare with frequency-domain based systems, which modifying the transformed coefficients of an image, LSB of the image pixels allow less modification than transformed coefficients before a perceptually visible distortion is introduced to the image. As a result, frequency-domain based systems allow a watermark with larger size or larger number of different watermark to be embedded than spatial-domain based systems.

Chapter 3

A watermarking system

3.1 Implementation of a watermarking system

According to [12], in order to make the watermark robust, it must be placed in perceptually significant components of the frequency spectrum of the data. In previous watermarking techniques, the watermark is easy to be removed either intentionally or unintentionally. They are not robust to common signal and geometric distortions. The reason is that these techniques do not explicitly place the watermark on the perceptually significant region of the data. In fact, they even try to avoid identifying the perceptually significant region as the destination of watermark because modification of these regions is prone to cause perceptual distortions.

In [12], it discussed the major challenge of placing watermark on perceptually significant region. That is, how to insert a watermark without causing visible distortions. It proposed an idea of spread spectrum coding of watermark. This idea is derived from spread spectrum communication technique. In which the

frequency domain of the image is viewed as a communication channel, the watermark is viewed as the signal that has to be transmitted through the channel, and the distortions are viewed as noise in the channel. The watermark inserted should be immune to those noises.

In spread spectrum communications, the energy of transmitted narrowband signal is spread over a larger bandwidth channel. Thus the signal energy in any single frequency is small and imperceptible. Similarly, in digital watermarking, the energy of the watermark is also spread over many frequency bins such that the energy in any one bin is undetectable. In this case, the watermark added is spread and any attempts of destroying the watermark will have to add noise to all frequency bins. It will cause severe degradation of the data before the watermark is destroyed.

The detection of watermark is done by extracting the watermark from the tested data, and then followed by a correlation test between the extracted watermark and the original watermark. Because the location of the watermark in the data and the algorithm of adding watermark are known, the extraction of watermark can be done in a trivial way.

3.1.1 Embedding watermark

A basic digital watermarking system is implemented in this research in order to perform testing. The part of watermark embedding is summarized in figure 3.1. There are many schemes of embedding watermark proposed. One of the possible scheme is wavelet based watermarking scheme [3][5][18]. In the watermarking system implemented here, similar to those wavelet based schemes, the original

image is first undergone a wavelet transform. [1] After that, a set of wavelet coefficients of the image can be obtained. The watermark sequence can be added into these wavelet coefficients. According to [12], the watermark should be explicitly added to the significant components of the signal. Therefore, the watermark sequence is now added to the large wavelet coefficients. In this implementation, a threshold is set as 1000. For the coefficients' absolute magnitude is larger than this threshold, watermark is added on it.

The form of the watermark sequence is a PN sequence. The PN sequence used is a positive or negative sequence. [2] The choice of magnitude of the watermark sequence is important. The watermark should be both robust and transparent. The smaller the magnitude of the watermark sequence, the more transparent but less robust. The larger the magnitude of the watermark sequence, the less transparent but more robust. As a result, a balance point has to be chosen in order to make the watermark both robust and transparent. The magnitude of the watermark sequence can be determined by try and error method. The magnitude of the watermark sequence is chosen to be 10 or -10. In short, for the absolute of the wavelet coefficients larger than 1000, either 10 or -10 is added on it, depending on the PN sequence generated.

For the capacity issue, it depends on the watermarking algorithm. However, this thesis is not focus on the issue of how to increase the capacity of the watermark. But rather the defense against geometric distortions mentioned above. Therefore, no special algorithm is used to increase the bit rate of data embedded. In my implementation, the length of the watermark sequence is depending on

[1] The source code is from http://www.cs.dartmouth.edu/ gdavis/wavelet/wavelet.html
[2] The source code is from http://www.sakitama.or.jp/cepstrum/2p-mseq.htm

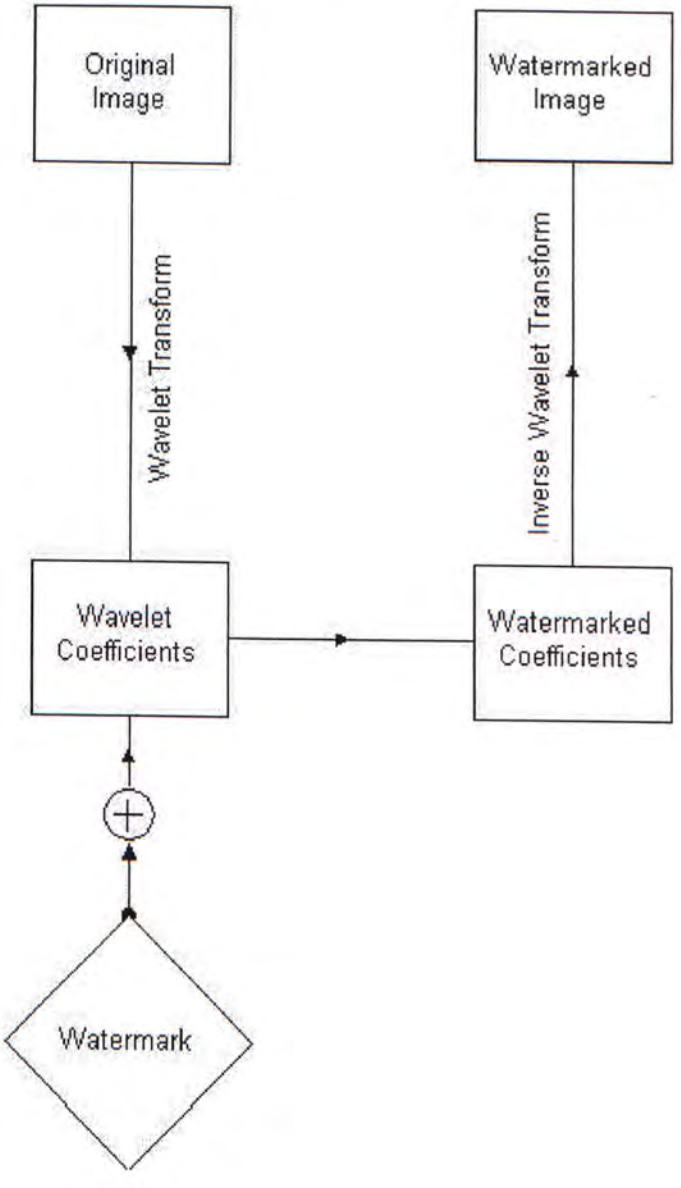

Figure 3.1: A watermarking system – Watermark Embedding

Figure 3.2: The test image before watermark embedding. The test image is a 512 by 512 pixel gray scale image.

threshold of wavelet coefficients to add watermark chosen. When the threshold is chosen as 1000, the length of watermark embedded is ranged from 1800 to 2300 bits for 512 by 512 gray scale image.

After watermark sequence is embedded into the wavelet coefficients, they are converted back to a watermarked image by inverse wavelet transform. From the tested images shown in figure 3.2 and figure 3.3, embedding watermark into the image does not introduce any visible distortion on the image.

3.1.2 Detecting watermark

The basic watermarking system for detecting watermark implemented in this research can be summarized by figure 3.4. The extraction of watermark from the test image is done by comparing the test image and the original image.

Figure 3.3: The test image after watermark embedding.

Both the test image and original image are undergone a wavelet transform. The difference between the wavelet coefficients of two images is regarded as the extracted watermark. This extracted watermark is then compared with the original watermark by a correlation test. The equation of correlation is

$$\frac{X \cdot \bar{X}}{\sqrt{(X \cdot X) \times (\bar{X} \cdot \bar{X})}} \tag{3.1}$$

where X is the original watermark sequence, $\bar{X}$ is the extracted watermark sequence. The operation '$\cdot$' is inner product.

The watermark sequence is a pseudo random number sequence. When two random sequences undergo a correlation, the result will be zero if the two sequences have no relation, one if two sequences are positive related, negative one if two sequences are negative related. In the watermarking detection, if a valid watermark is detected, the correlation value between two watermark sequences

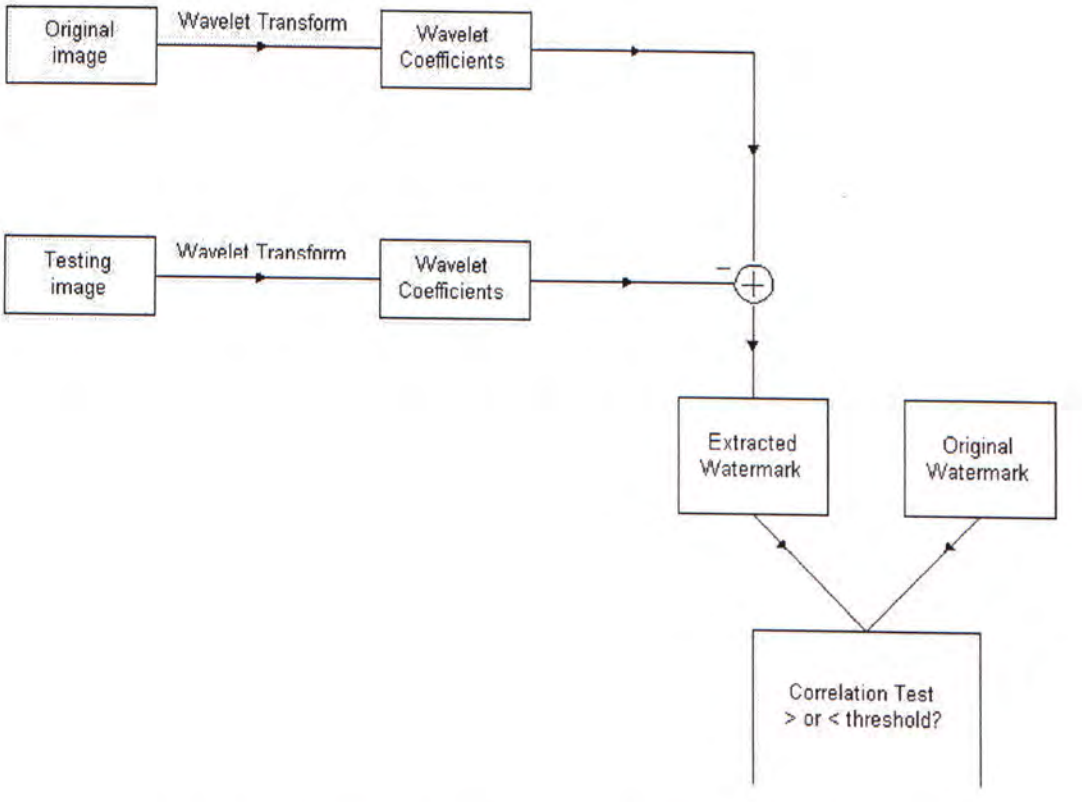

Figure 3.4: A watermarking system – Watermark Detection

should be close to one. If an invalid watermark is detected, the correlation value should be close to zero.

Tests are done based on this implementation. An unmodified watermarked image is undergone a watermark detection. The correlation value between the extracted watermark sequence and the original watermark sequence is 0.997, which is very close to one. Sixteen other random chosen images are passed to this correlation test. The results are shown in table 3.1.

Unmodified watermarked image	0.997
Unmarked image	0.00
Random chosen image 1	-0.03
Random chosen image 2	-0.04
Random chosen image 3	-0.02
Random chosen image 4	-0.04
Random chosen image 5	-0.05
Random chosen image 6	-0.03
Random chosen image 7	-0.06
Random chosen image 8	-0.04
Random chosen image 9	-0.04
Random chosen image 10	-0.01
Random chosen image 11	-0.001
Random chosen image 12	-0.06
Random chosen image 13	-0.007
Random chosen image 14	-0.05
Random chosen image 15	-0.02
Random chosen image 16	-0.04

Table 3.1: The correlation value of unmodified watermarked image and other random chosen images

From the results, the random chosen images give very low values in correlation tests when comparing with the unmodified watermarked image. Thus the threshold of the correlation tested is set to be 0.1. In other words, if the output

in correlation test is higher than 0.1, watermark is considered to be existing in the image.

3.2 Robustness testing on the watermarking system

According to [12], in order to make a watermark robust, it must be placed in the perceptually significant region of an image. In the implementation of the research, the watermark sequence is placed on the wavelet coefficients that are larger than the threshold. The watermark is now placed in the significant region and it is now considered to be robust to the attacks mentioned in [12]. Tests are done on the attacks of JPG compression, noise adding, cropping and collision. The results are shown in table 3.2. It is found that the watermarking system implemented is robust to all the tests done. For the distortion such as JPG compression and noise adding, the correlation value is still above the threshold when the distortion is so large that a visible degradation on the image quality appeared. As a result, it is concluded that this watermark is robust to these common signal processing and collision attacks.

3.3 Geometric attacks to the watermark system

Beside common signal processing and collision attacks, geometric distortion is also a common attack to a watermarked image. Since combating geometric

Test image: a gray scale image, size if 512 by 512 pixels.
The correlation value for an unmodified marked image: 0.997
JPG compression: Compression factor=59 in paint shop pro 5.0: 0.8856 Compression factor=79 in paint shop pro 5.0: 0.6895 Compression factor=99 in paint shop pro 5.0: 0.1381
Noise adding: Noise amount=10 in photo shop 5.0: 0.861 Noise amount=20 in photo shop 5.0: 0.652 Noise amount=70 in photo shop 5.0: 0.135
Cropping: Crop a 370 by 370 section from the marked image: 0.793 Crop a 270 by 270 section from the marked image: 0.607
Collision: Averaging a marked and an unmarked image: 0.3601

Table 3.2: The test results of various kinds of attacks

distortions is the focus of this thesis. A detailed description of the geometric distortions and their effect on watermark is done in this section, with the plotting of correlation value vs the magnitude of distortion. Moreover, the method of implementing a program to do this geometric distortion is also discussed.

3.3.1 The three distortions: Translation, Dilation and Rotation

1. **Translation**

   Suppose $f(x, y)$ is an image, a translated (or shifted) image with α and β shifted pixel in x and y directions can be expressed as

$$\mathcal{T}_{\alpha,\beta}\big(f(x,y)\big) = f(x+\alpha, y+\beta) \tag{3.2}$$

where $\mathcal{T}_{\alpha,\beta}$ is the operator of translation with α and β pixels translated in x and y directions respectively. The relation between the number of pixel translated in a watermarked image and the correlation test value in the watermarking detection is shown in figure 3.5.

2. **Dilation**

Suppose $f(x,y)$ is an image, a dilated (or rescaled) image with the dilation scale factor ρ can be expressed as

$$\mathcal{D}_{\rho}\big(f(x,y)\big) = f(\rho x, \rho y) \tag{3.3}$$

where $\mathcal{D}_{\rho}$ is the operator of dilation with dilation factor ρ. With the help of logarithmic-polar coordinate, dilation can also be expressed as translation. Consider the transformation from x-y coordinate to logarithmic-polar coordinate, suppose there is a point $(x,y) \in \mathcal{R}$, define:

$$\begin{aligned} x &= e^{r} cos\ t \\ y &= e^{r} sin\ t \end{aligned} \tag{3.4}$$

where $r \in \mathcal{R}$ and $0 < t < 2\pi$. For every point (x,y), there is a point (r,t) that uniquely corresponds to it and $\tilde{f}(r,t)$ is another representation of $f(x,y)$. After that, dilation is converted into translation in the r and t coordinate.

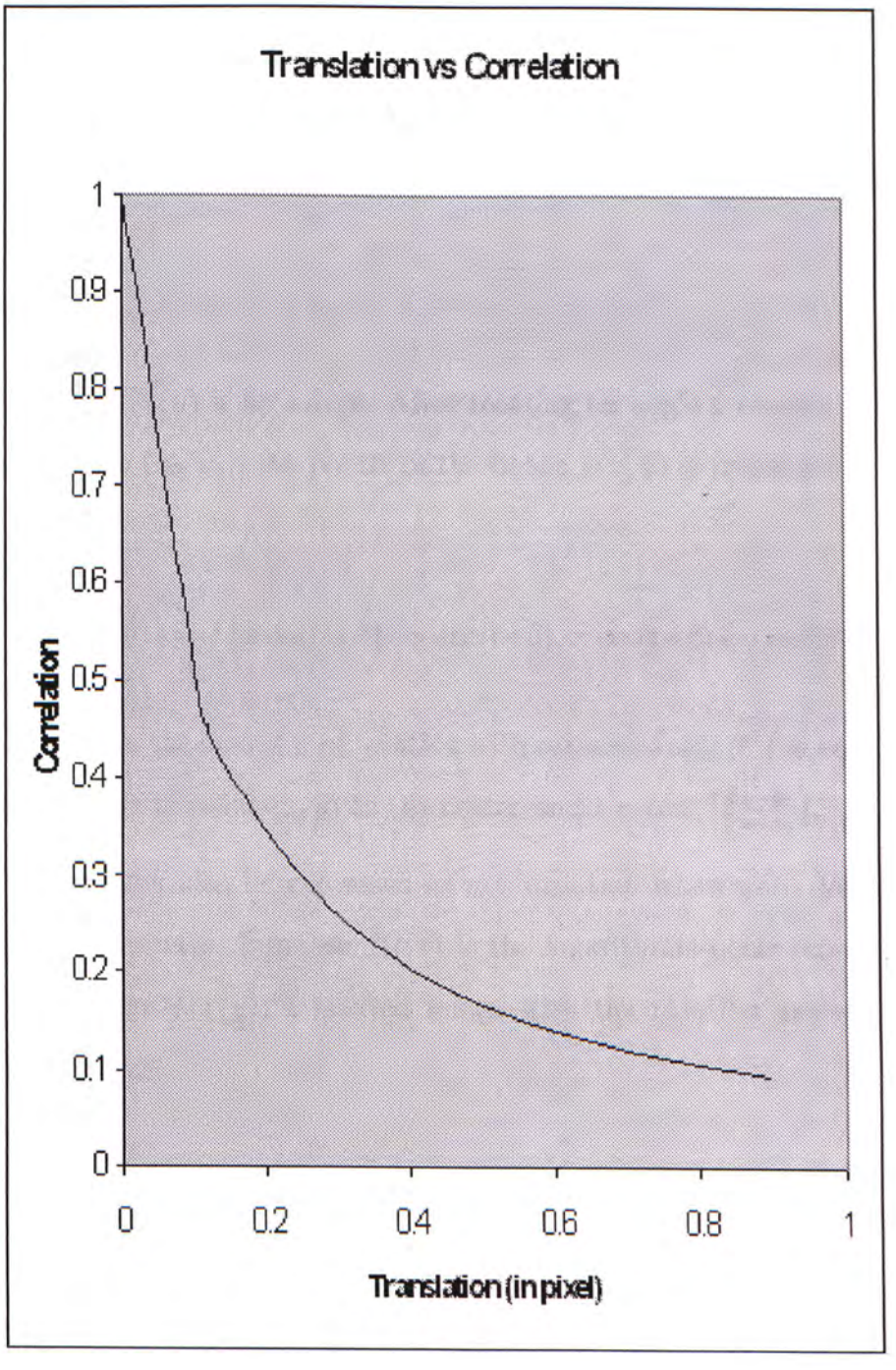

Figure 3.5: Translation vs Correlation

$$\mathcal{D}_\rho\big(\tilde{f}(r,t)\big) = \tilde{f}(r + ln\ \rho, t) \tag{3.5}$$

Similar to translation, the relation between dilation of a watermarked image and the correlation test value in watermark detection is shown in figure 3.6.

3. **Rotation**

Suppose $f(x,y)$ is an image. After rotating an angle θ around the center of rotation (x_0, y_0), the points in the image $f(x,y)$ is transformed to

$$\mathcal{R}_\theta\big(f(x,y)\big) = f\big(x\ cos(t+\theta) - y\ sin(t+\theta), x\ sin(t+\theta) + y\ cos(t+\theta)\big) \tag{3.6}$$

where $\mathcal{R}_\theta$ is the operator of rotation with rotation angle θ. t in equation 3.6 is the angle of point (x,y) to the center and $t = tan^{-1}\big(\frac{y-y_0}{x-x_0}\big)$.

Rotation can also be expressed as a translation when using logarithmic-polar coordinate. Suppose $\tilde{f}(r,t)$ is the logarithmic-polar representation of an image $f(x,y)$, a rotated image with the rotation angle θ can be expressed as

$$\mathcal{R}_\theta\big(\tilde{f}(r,t)\big) = \tilde{f}(r, t+\theta)$$

A graph showing the relation between rotation of a watermarked image and the correlation test value in watermark detection is shown in figure 3.7.

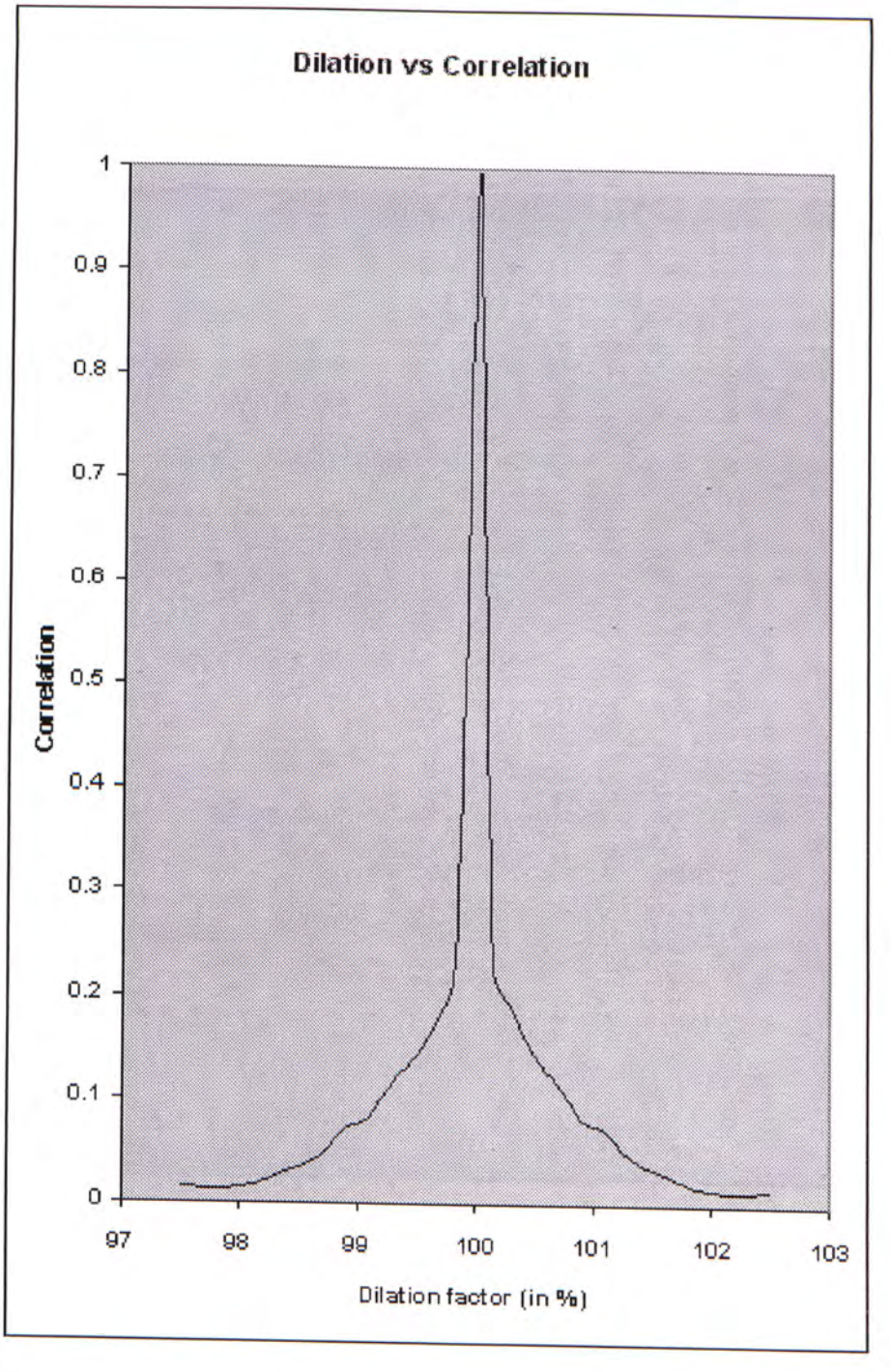

Figure 3.6: Dilation vs Correlation

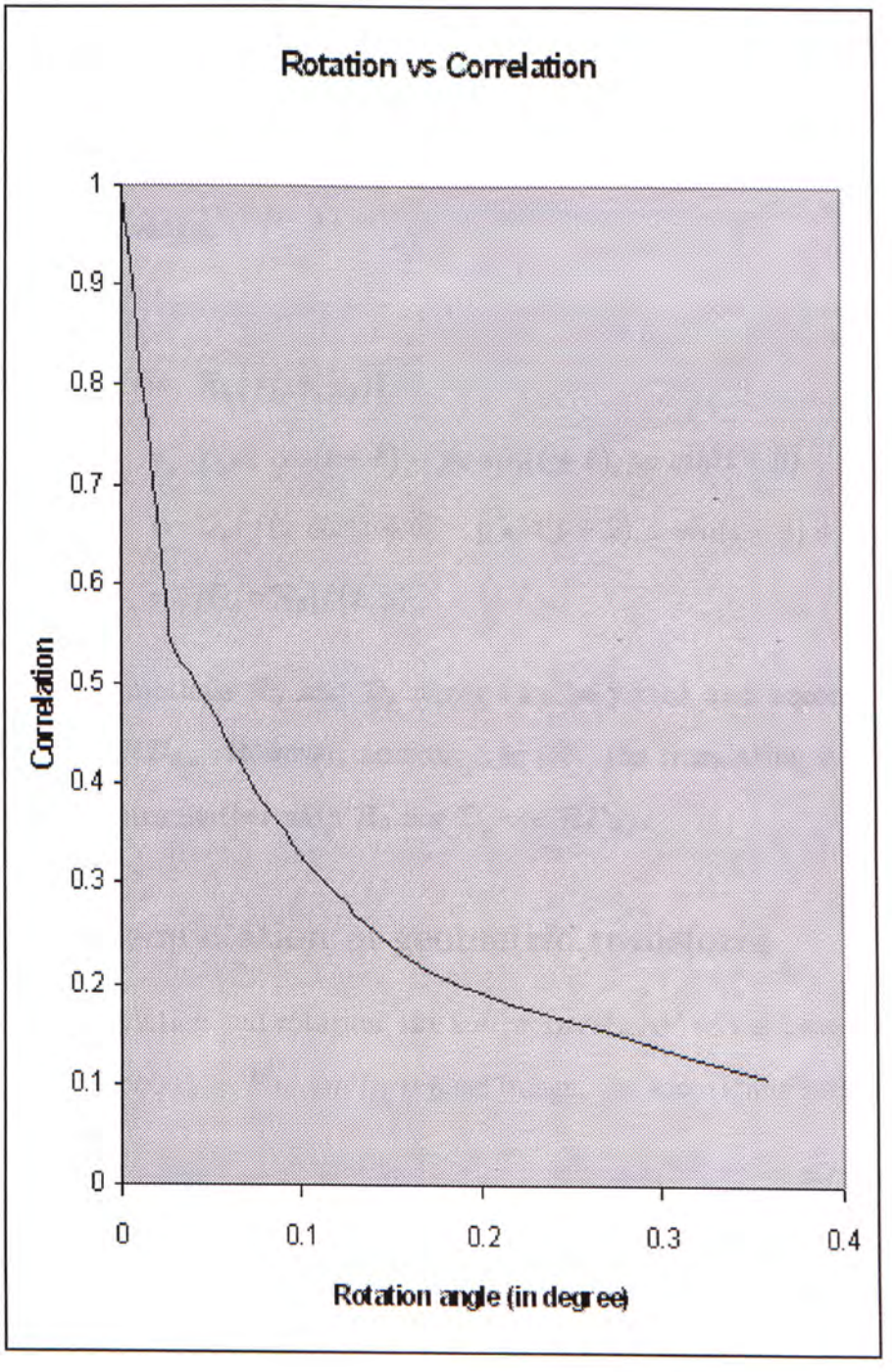

Figure 3.7: Rotation vs Correlation

3.3.2 The commutative property of rotation and dilation

The transformations $\mathcal{R}_\theta$ and $\mathcal{D}_\rho$ mentioned above have commutative property. In other words, doing rotation and then dilation on an image has the same effect as doing dilation and then rotation on that image [23]. It can be shown by following derivation.

$$\begin{aligned}
[\mathcal{R}_\theta \circ \mathcal{D}_\rho] f(x,y) &= \mathcal{R}_\theta\big(f(\rho x, \rho y)\big) \\
&= f(\rho x\ cos(t+\theta) - \rho y\ sin(t+\theta), \rho x\ sin(t+\theta) + \rho y\ cos(t+\theta)) \\
&= \mathcal{D}_\rho\big(f(x\ cos(t+\theta) - y\ sin(t+\theta), x\ sin(t+\theta) + y\ cos(t+\theta))\big) \\
&= [\mathcal{D}_\rho \circ \mathcal{R}_\theta] f(x,y) \qquad (3.7)
\end{aligned}$$

The transformations $\mathcal{R}_\theta$ and $\mathcal{D}_\rho$ above can be joined and become a joint transformation $\mathcal{RD}_{\theta,\rho}$. However, according to [25], the translating transformation $\mathcal{T}_{\alpha,\beta}$ commutes neither with $\mathcal{R}_\theta$ nor $\mathcal{D}_\rho$ nor $\mathcal{RD}_{\theta,\rho}$.

3.3.3 Implementation of geometric transform

In translation, dilation and rotation, the transformed pixel values have to be calculated by interpolation. For a m by n pixel image, the equation of interpolation is

$$f(\hat{x},\hat{y}) = \sum_{i=1}^{m}\sum_{j=1}^{n} f(i,j) sinc(\hat{x}-i, \hat{y}-j)$$

where $f(\hat{x},\hat{y})$ is the transformed pixel values. $(\hat{x},\hat{y})$ has different relationship with (x,y) for different transformation. For translation:

$$\hat{x} = x + \alpha$$

$$\hat{y} = y + \beta$$

For dilation:

$$\hat{x} = \rho x$$

$$\hat{y} = \rho y$$

For rotation:

$$\hat{x} = x\ cos(t + \theta) - y\ sin(t + \theta)$$

$$\hat{y} = x\ sin(t + \theta) + y\ cos(t + \theta)$$

After the new pixel position $(\hat{x}, \hat{y})$ is calculated, the values of transformed pixels can be found by interpolation.

Sinc function is used in interpolation calculation. In this way, all the pixel values in an image to find out a transformed pixel value. However, the complexity of calculation of each pixel is high. For each pixel, it needs m by n number of addition and multiplication. In calculating the whole image pixels, the number of addition and multiplication is $(m \times n)^2$, which makes the calculation time-consuming. In order to reduce the complexity, a simplified function is used instead of sinc function. The simplified function is indeed the approximated version of sinc function. Its tail is not extended to infinity and drops to zero in a limited interval. Then, not all the pixel values in an image are

needed to find out a transformed pixel value, but rather the pixels in a limited region beside the transformed pixel are needed. This can reduce the complexity in translating, dilating and rotating calculations. But this calculation is more lossy than using sinc function, which means that information is lost in interpolation and a transformed image cannot recover exactly the original image even if it is transformed back.

Chapter 4

General Defense on Geometric Distortions

There is a general method to deal with geometric distortions as shown in the literature. It is to design special watermark that resistant to these distortions [24][23]. In their papers, they all claim that the watermarking schemes proposed are resistant to all three geometric distortions: translation, dilation and rotation.

There is another way to deal with geometric distortion, that is distortion correction. Up to our knowledge, it is not been discussed in the literature much. The main idea is easy: detect the distortion magnitude and try to correct the distortion by translate back, dilate back or rotate back. The two distortion detection techniques mentioned in section 4.2 are easy to understand but more erroneous. They are done either by brute force or by try-and-error. In the next chapter, phase angle comparison is used in calculating the distortion level, which yields a less erroneous result.

4.1 Special designed watermark

4.1.1 Ring-shaped watermark

The idea of ring-shaped watermark proposed in [24]. It is a watermarking system based on discrete Fourier transform (DFT). A watermark signal, W, is added to the DFT coefficients, M, of an image. It is same as the watermark embedding shown in chapter three. The operation of watermark embedding is summarized in equation 4.1, where $\bar{M}$ is the DFT coefficients after embedding watermark. It will be inversely transformed into a watermarked image. a is the magnitude of the watermark.

$$\bar{M} = M + aW \tag{4.1}$$

The difference between this watermarking scheme and the ordinary watermarking schemes is on the property of the shape of the watermark. It is a two dimensional zero-mean random bi-valued sequence covering on M. It is divided into S ring-shaped sectors and these rings are homocentric with the centers on the middle frequency of M. For each circular sector of W, the same value 1 or -1 is assigned. As a result, the actual watermark embedded on M is a ring-shaped watermark with either a or $-a$ on each ring.

The detection of this ring-shaped watermark is same as the watermark detection mentioned on previous chapter, which is by correlation test. This watermarking scheme is resistant to rotation because the rotation in spatial domain results a rotation of coefficients in Fourier domain by same degree (section 4.1.3, the rotation property). Since the watermark is divided into rings and each ring

has identical value, rotation on the image does not affect the watermark embedded.

4.1.2 Phase Taylor invariance

Phase Taylor invariance is an idea proposed in [25]. It is started with a two dimensional Fourier transform of $f(x, y)$, where $f(x, y)$ can be a function of an image.

$$F(\omega_x, \omega_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)e^{-j(x\omega_x + y\omega_y)}dxdy$$

The phase Taylor invariance is defined to be

$$T(\omega_x, \omega_y) = F(\omega_x, \omega_y)e^{-j(a\omega_x + b\omega_y)} \tag{4.2}$$

where

$$\begin{aligned} a &= -\frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} xf(x, y)dxdy}{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)dxdy} \\ b &= -\frac{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} yf(x, y)dxdy}{\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)dxdy} \end{aligned} \tag{4.3}$$

The point (a, b) from equation 4.3 is the center of gravity of the function $f(x, y)$. And the invariance $T(\omega_x, \omega_y)$ in equation 4.2 is equivalent to a translation of the origin of the function $f(x, y)$ to (a, b), its center of gravity. This $T(\omega_x, \omega_y)$ is translation invariant, ie, the T function of a translated image is the same as that of the image before translation. This can be proved intuitively. A translated image should have the center of gravity be translated in the same amount. The location of center of gravity related to other image pixels is unchanged. As a result, translating the image to its center of gravity results in

same output for an image before and after translation. Since phase Taylor invariance is invariant under translation, a translation invariant watermark can be done by embedding and extracting watermark on $T(\omega_x, \omega_y)$.

However, this phase Taylor invariance has its drawback. The watermark inserted on it is only robust for translation, but not for other common signal processing such as JPG compression and decompression, resampling and requantization etc or any lossy interpolation performed in geometric distortions. Because all these processes can alter the center of gravity of an image. The transformation in equation 4.2 depends on an accurate calculation of center of gravity. Any small alteration on the center of gravity can result on a different output from the transformation and thus the watermark embedded cannot be detected anymore. Simulation results show that phase Taylor invariance performs poor in the processes which can alter the center of gravity. For instance, once the image is translated using lossy interpolation which can alter the center of gravity, the correlation result in watermark detection drops below the threshold. Therefore, it is concluded that phase Taylor invariance can only combat translation theoretically.

4.1.3 Rotation, dilation and translation invariant watermark

The idea of rotation, dilation and translation invariant watermark is proposed in [23]. Its main idea is to transform an image to a domain that is invariant to these three distortions. Both embedding and extracting of watermark are done in this domain. Its idea is same as placing watermark in phase Taylor invariant.

But it is resistant to translation as well as rotation and dilation.

Before discussing the transform of image to this rotation, dilation and translation invariance, properties about Fourier transform have to be discussed first. The frequency domain and the spatial domain form a dual relation. Any transform in spatial domain has its corresponding transform in frequency domain. In the following properties, $f(x, y)$ is a function in spatial domain and $F(\omega_x, \omega_y)$ is its Fourier transform.

- **Property 4.1.1** *Translation property*

  A translation in the spatial domain causes a linear shift in the phase component of $F(\omega_x, \omega_y)$.

$$f(x + \alpha, y + \beta) \leftrightarrow F(\omega_x, \omega_y) e^{-j2\pi(\omega_x \alpha + \omega_y \beta)}$$

- **Property 4.1.2** *The reciprocal scaling property*

  Dilation in spatial domain causes an inverse dilation in the frequency domain.

$$f(\rho x, \rho y) \leftrightarrow \frac{1}{\rho} F\left(\frac{\omega_x}{\rho}, \frac{\omega_y}{\rho}\right)$$

- **Property 4.1.3** *The rotation property*

  Rotation an image in its spatial domain results a same angle rotation in its frequency domain. This property has also been used in ring-shaped watermark. (section 4.1.1)

$$f(x\ cos\theta - y\ sin\theta, x\ sin\theta + y\ cos\theta) \leftrightarrow F(\omega_x cos\theta - \omega_y sin\theta, \omega_x sin\theta + \omega_y cos\theta)$$

- **Property 4.1.4** *Translation invariance property*

  Consider the translation property for Fourier transform, a spatial translation only affect the phase representation of an image. The magnitude of Fourier transform is not affected. As a result, Fourier transform magnitude is translation invariant.

- **Property 4.1.5** *Rotation and dilation invariance property*

  With the representation of logarithmic-polar coordinate, rotation and dilation can be expressed as translation (see section 3.3). By translation invariance property, translation is invariant under Fourier transform. Therefore, it can be concluded that rotation and dilation are also invariant under Fourier transform in logarithmic-polar coordinate. The Fourier transform in this situation is called Fourier Mellin transform [23], which will be discussed in details next chapter.

For the rotation, dilation and translation invariance, consider two operations: $\mathcal{F}$ and $\mathcal{F}_{\mathcal{M}}$, where $\mathcal{F}$ is the Fourier transform operation and $\mathcal{F}_{\mathcal{M}}$ is the Fourier Mellin transform operation. They join together to be a hybrid operator $\mathcal{F} \circ \mathcal{F}_{\mathcal{M}}$. Apply this hybrid operator to an image $f(x, y)$

$$I_1 = [\mathcal{F} \circ \mathcal{F}_{\mathcal{M}}] f(x, y)$$

I_1 in the above operation is the domain that rotation, dilation and translation invariant. Now, apply the hybrid operator to a rotated, dilated and translated image.

$$I_2 = [\mathcal{F}_{\mathcal{M}} \circ \mathcal{F} \circ \mathcal{R}_\theta \circ \mathcal{D}_\rho \circ \mathcal{T}_{\alpha,\beta}] f(x, y)$$

By the rotation property (property 4.1.3), the priority of rotation and Fourier transform can be exchanged.

$$I_2 = [\mathcal{F}_{\mathcal{M}} \circ \mathcal{R}_\theta \circ \mathcal{F} \circ \mathcal{D}_\rho \circ \mathcal{T}_{\alpha,\beta}]f(x,y)$$

By the reciprocal scaling property (property 4.1.2), dilation operator and Fourier transform operator can also exchange priority with the dilation factor inverted.

$$I_2 = [\mathcal{F}_{\mathcal{M}} \circ \mathcal{R}_\theta \circ \mathcal{D}_{\frac{1}{\rho}} \circ \mathcal{F} \circ \mathcal{T}_{\alpha,\beta}]f(x,y)$$

Apply the translation invariance property here (property 4.1.4), the equation above can be contracted to

$$I_2 = [\mathcal{F}_{\mathcal{M}} \circ \mathcal{R}_\theta \circ \mathcal{D}_{\frac{1}{\rho}} \circ \mathcal{F}]f(x,y)$$

Similarly, the equation above can be further contracted by using rotation and dilation invariance property (property 4.1.5).

$$I_2 = [\mathcal{F}_{\mathcal{M}} \circ \mathcal{F}]f(x,y)$$

Hence, $I_1 = I_2$ and therefore, I_1 or I_2 is the domain invariant under rotation, dilation and translation. The watermark embedded in this domain is also invariant under rotation, dilation and translation. In [23], it has an example on embedding and detecting digital watermark under this invariance.

4.2 Distortion detection

The idea of distortion detection is to detect the distortion level and then recover the image by doing correction on it. The correction is simply an inverse

transform on the image. The idea mentioned in section 4.1 is to design a new algorithm of embedding and retrieving watermark in order to make their watermark scheme robust to geometric distortions. Compare with the idea above, the advantage of distortion detection is that watermark embedding and retrieving algorithm need not be changed. The algorithms that proven to be robust to different kinds of distortions (eg. the watermarking scheme in [12]) can still be used. An additional robustness against geometric distortions can be gained on these existing algorithms.

There are two distortion detection methods listed below. The brute-force method is for detecting the amount of translation. And the interactive method is for dealing with cropping and dilation. However, the brute-force method is proven by the simulation results to be not accurate enough and not feasible. In the next chapter, another method called phase angle comparison will be used. It can be used in translation detection. With the help of logarithmic-polar coordinate and Fourier Mellin transform, dilation and rotation can also be dealt with distortion detection. Moreover, phase angle comparison produces a more accurate result on distortion detection, as shown by experiment data.

4.2.1 Brute-force method

Brute-force method tries to detect the distortion by try and error. This method can be used in detecting translation. Its advantages are easy to understand and no theory needed. However, this method has large experimental error.

The method is described as following. A small image segment is cropped from a translated image and the original image respectively. The image segment from the original image is shifted into different levels and then compare with

the translated image. The method of comparing is by correlation test. The translation level which produce the highest correlation level is considered to be the detected translation value.

The results of translation detection by brute-force method is shown in the table 4.1. From the table, it is found that the brute-force method has a large detection error. Its error can be as high as 60% when detecting small translation value. The phase angle comparison method in the next chapter yields a much better translation detection result than brute-force method.

Translated value (number of pixel)	Detected shifted value (number of pixel)	Percent of error
0.10	0.04	60.0%
0.11	0.04	63.6%
0.12	0.05	58.3%
0.13	0.06	53.8%
0.14	0.07	50.0%
0.15	0.08	46.7%
0.16	0.09	43.8%
0.18	0.11	38.9%
0.20	0.13	35.0%
0.22	0.16	27.3%
0.24	0.17	29.2%
0.26	0.20	23.1%
0.30	0.24	20.0%
0.40	0.35	12.5%
0.50	0.44	12.0%
0.60	0.54	10.0%
0.70	0.66	5.7%
0.80	0.75	6.3%
0.90	0.84	6.0%

Table 4.1: The detection of translation

Figure 4.1: The test image in example one.

4.2.2 Interactive method

The distortions such as cropping and dilation can be detected by interactive method. Its main concept is detecting the distortion by human eye in the first pace. Then the detection is fine tuned by computer and find out the optimal value of the detection.

To test whether a cropped and dilated image has watermark, its dilation factor and the cropping location have to be determined first. The cropped and dilated image segment is first adjusted to its original size approximately by human eyes. It can be done by recognizing the distance between the edges or corners of the objects inside the image. The exact adjustment of the size of the segment and the exact location of the segment within the whole image are then fine tuned by computer programs. The fine tuning is by try-and-error technique. Two examples are shown below.

- **Example one**

  The image segment shown in figure 4.1 is cut out from a watermarked

image, then dilated to a size of 325 by 342 pixel. The interactive method in detecting watermark is summarized into these steps:

Step 1: Adjust the image segment in figure 4.1 to a size that close to its original size. This adjustment is done by recognizing the distance between objects in the image by human eye.

Step 2: The scaled test image is wrapped around the original image and find out the difference value between the pixels of test image segment and the original image. The difference value calculation is

$$\text{difference value} = \sum_{i=1}^{n} \Big(\text{abs}(x_i - \bar{x}_i)\Big) \tag{4.4}$$

where n is number of pixels in the scaled test image, abs is absolute value, x_i is the value of the i-th pixel in the original image and $\bar{x}_i$ is the value of the i-th pixel in the scaled test image segment.

Step 3: The test image is scaled to other sizes which are slightly larger or smaller than the size in **Step 1**. Then repeat **Step 2**. Different combinations of rescaled size and wrapping location of the test image segment yield different outputs from equation 4.4. The combination with the minimum difference value is the combination that closest to the original image, which is the desirable rescaled size and wrapping location of the test image. Table 4.2 shows the result of the detection.

Step 4: Replace an unmarked image by the rescaled test image segment at the desirable wrapping location. And then pass it to watermark detection test.

Dilate the image segment to (size in pixel)	Wrapping location	Difference value
430 by 452	(26,33) to (455,484)	1530667
431 by 454	(24,33) to (454,486)	1228830
432 by 455	(24,33) to (455,487)	1035773
433 by 456	(23,32) to (455,487)	950791
434 by 457	(23,32) to (456,488)	932217
435 by 458	(22,31) to (456,488)	1167419
436 by 459	(22,31) to (457,489)	1335581
437 by 460	(21,31) to (457,490)	1593229
438 by 461	(21,30) to (458,490)	1803889
439 by 462	(20,30) to (458,491)	2036820

Table 4.2: The results of difference value in example one.

From the table 4.2, the minimum difference value can be obtained by dilating the image segment to a size of 434 by 457 and compare it with the pixels (23,32) to (456,488) of the original image. Therefore, it can be concluded that the image segment is a dilated segment that cut from (23,32) to (456,488) of the whole image.

After the dilation factor and the location of the image segment are determined, watermark detection can be performed. It is done by scaling the test image segment to a size of 434 by 457 pixels, then replacing (23,32) to (456,488) of an unmarked image by this enlarged image segment. The image after replacement is shown in figure 4.2. This image is then undergone a watermark extraction and correlation test. The test result is 0.1167. This correlation value obtained is low comparing with correlation value of a unmodified watermarked image (0.997). It is because the test image in figure 4.1 is cropped and scaled to a smaller size. Information of

Figure 4.2: The image after replacement in example one.

watermark lost in these operations and cause the correlation value lowered. But this correlation value is still above the threshold, which is 0.1. As a result, we can still conclude that this image segment has watermark.

- **Example two**

  A piece of image segment is cut out from an image without watermark and then scaled. It is shown in figure 4.3. It is now undergone watermark detection by interactive method. Similar to the four steps in example one, a test on the calculation of difference value will be performed in order to find out which combination of dilation factor and wrapping location is the desirable one. Table 4.3 shows the test result.

  From the test results, it can be concluded that the image segment is a dilated image segment cut from (32,52) to (432,459) of the whole image. To

Figure 4.3: The test image in example two.

perform watermark detection, this image segment is scaled to a size of 401 by 408 pixels. Then the pixels in (32,52) to (432,459) of an unmarked image is replaced by this enlarged image segment. The image after replacement is shown in figure 4.4, and is then passed to watermark extraction and correlation test. The test result is -0.0036, which is below the threshold. As a result, it is concluded that the image segment shown in figure 4.3 does not contain watermark.

Dilate the image segment to (size in pixel)	Wrapping location	Difference value
395 by 402	(35,43) to (429,444)	1655834
396 by 403	(35,53) to (430,455)	1512171
397 by 404	(34,53) to (430,456)	1307004
398 by 405	(33,53) to (430,457)	1114662
399 by 406	(33,52) to (431,457)	954076
400 by 407	(32,52) to (431,458)	647742
401 by 408	(32,52) to (432,459)	591489
402 by 409	(31,51) to (432,459)	651384
403 by 410	(31,51) to (433,460)	839668
404 by 411	(30,51) to (433,461)	1108705
405 by 412	(30,50) to (434,461)	1325991

Table 4.3: The results of difference value in example two.

Figure 4.4: The image after replacement in example two.

Chapter 5

Specific Defense in Geometric Distortions - Phase angle comparison

The distortion detection and correction can be further improved by the technique of phase angle comparison. The idea of phase angle comparison is to transform the original image and the distorted image to its frequency domain. By comparing the phase angles of two transformed images, the level of distortion can be detected. Phase angle comparison technique can be used to detect small distortion. For large distortion, it has to be detected by other methods described in further work. The three sections below demonstrate the detection of three distortions by phase angle comparison.

5.1 Translation Detection

From the previous chapter, it was found that the brute-force method of distortion detection and phase Taylor invariance does not combat translation in a satisfactory way. Phase angle comparison is another distortion detection technique which is theoretically feasible. Moreover, it is concluded from the experimental results that phase angle comparison is a method that yields less erroneous results.

First of all, consider a one-dimensional case. Let $f(x)$ be a signal and $f_t(x)$ be the translated version of this signal, ie, $f_t(x) = \mathcal{T}_\alpha\big(f(x)\big) = f(x+\alpha)$, where α is the magnitude of translation. The Fourier transform of $f(x)$ is

$$F(\omega) = \int_{-\infty}^{\infty} f(x)e^{-jx\omega}dx$$

From the translation property of Fourier transform (property 4.1.1 in section 4.1.3)

$$f(x+\alpha) \rightarrow e^{j2\pi\omega\alpha}F(\omega)$$

Then

$$F_t(\omega) = e^{j2\pi\omega\alpha}F(\omega)$$

$$\Rightarrow \frac{F_t(\omega)}{F(\omega)} = e^{j2\pi\omega\alpha}$$

Let

$$G_t(\omega) = \angle\frac{F_t(\omega)}{F(\omega)} \tag{5.1}$$

where $\angle$ is a phase angle symbol. The plot of $G_t(\omega)$ is a straight line with the slope $2\pi\alpha$. From the slope, the amount of translation α can be found. Moreover, the above derivation can be extended to two-dimensional space in order to find the translation magnitude in x and y directions.

The results of translation detection by phase angle comparison are shown in table 5.1. From the results, it is found that this detection method yields a better result than the brute-force method mentioned before. The error is not more than 2%.

Translated value (number of pixel)	Detected translation value (number of pixel)	Percent of error
0.10	0.0982	1.8%
0.11	0.1080	1.8%
0.12	0.1178	1.8%
0.13	0.1276	1.8%
0.14	0.1374	1.9%
0.15	0.1472	1.9%
0.16	0.1570	1.9%
0.18	0.1767	1.8%
0.20	0.1963	1.9%
0.22	0.2159	1.9%
0.24	0.2355	1.9%
0.26	0.2551	1.9%
0.30	0.2943	1.9%
0.40	0.3923	1.9%
0.50	0.4903	1.9%
0.60	0.5882	2.0%
0.70	0.6861	2.0%
0.80	0.7840	2.0%
0.90	0.8820	2.0%

Table 5.1: Translation detection using phase angle comparison.

After the distortion is detected, the image is translated back and passed to

watermark detection. The correlation values in watermark detection are restored to a value ranging from 0.2980 to 0.3127. Compare with the correlation value when the image is translated 0.9 pixel, which is 0.092, the restored correlation value after back translation is a large improvement. It restores the correlation value from a value below threshold to a value above threshold. However, for small translation distortion, the restored correlation value is even lower. For example, in translating the image 0.1 pixel. The correlation value after translation is 0.4633 and after back translation is 0.3107. It is because the lossy interpolation in doing translation introduces error and the effect of restoration is cancelled by the double interpolation in forward and backward translation.

For the implementation, it is straightforward. The environment used is MatLab. In MatLab, there is a function **FFT** to perform Fourier transform. After applying this function to an image pixel array, a Fourier coefficient array is produced. And then the phase comparison can be done by comparing the two arrays of Fourier coefficient directly.

5.2 Dilation Detection

From section 3.3, the logarithmic-polar coordinate shown in equation 3.4 demonstrated that dilation can be expressed as translation. After it is expressed as translation, the dilation factor can also be found by the phase angle comparison.

In the experiment, only the dilation factor is concerned. Therefore, the logarithmic-polar coordinate is reduced into one-dimensional space with r only. Represent a function $f(x)$ by logarithmic coordinate $\tilde{f}(r)$ by substituting $x = e^r$.

The Fourier transform of $\tilde{f}(r)$ is

$$\tilde{F}(\omega) = \int_{-\infty}^{\infty} \tilde{f}(r) e^{-jr\omega} dr$$

By changing variable, $\tilde{F}(\omega)$ can be found out directly by $f(x)$.

$$dr = \frac{1}{x} dx$$

Then

$$\begin{aligned} \tilde{F}(\omega) &= \int_{-\infty}^{\infty} \tilde{f}(r) e^{-jr\omega} dr \\ &= \int_{-\infty}^{\infty} f(x) e^{-j(lnx)\omega} \frac{1}{x} dx \\ &= \int_{-\infty}^{\infty} f(x) x^{-j\omega} \frac{1}{x} dx \end{aligned} \tag{5.2}$$

When using discrete calculation, summation ($\sum$) is used instead of integration. As a result, equation 5.2 is modified to

$$\tilde{F}(\omega) = \sum_{x} f(x) x^{-j\omega} \frac{1}{x} \tag{5.3}$$

Suppose the dilated version of $f(x)$ is $f_d(x)$, ie, $f_d(x) = \mathcal{D}_\rho\big(f(x)\big) = f(\rho x)$, where ρ is the rescaling factor. Expressing $f_d(x)$ in logarithmic coordinate:

$$f_d(x) = f(\rho x)$$

$$\begin{aligned} \Rightarrow f_d(e^r) &= f(\rho e^r) \\ &= f(e^{r+ln\ \rho}) \end{aligned}$$

Then

$$\tilde{f}_d(r) = \tilde{f}(r + ln\ \rho)$$

Similar to previous section, consider the Fourier transform of $f_d(x)$ and $f(x)$. We have,

$$\tilde{F}_d(\omega) = e^{j2\pi\omega ln\rho}\tilde{F}(\omega)$$

$$\Rightarrow \frac{\tilde{F}_d(\omega)}{\tilde{F}(\omega)} = e^{j2\pi\omega ln\rho}$$

Let

$$G_d(\omega) = \angle\frac{\tilde{F}_d(\omega)}{\tilde{F}(\omega)} \tag{5.4}$$

The plot of $G_d(\omega)$ is a straight line with slope $2\pi ln\rho$. From the slope of the curve, the rescaling factor can be found. The results of rescaling detection is shown in table 5.2.

Dilation factor (in percentage)	Detected dilation factor (in percentage)	Percent of error
110	112.5	2.3%
99	99.4	0.4%
97	96.2	0.8%
90	88.9	1.2%
85	85.8	0.9%
80	78.7	1.8%
70	72.2	3.1%

Table 5.2: Rescaling detection using logarithmic coordinate and phase angle comparison.

The result of back dilation is similar to back translation. The restored correlation values are ranging from 0.2231 to 0.3601. They are all above the threshold and the watermark is considered to be recovered.

The implementation of dilation detection is more complicated than that of translation detection. Because there is no function in MatLab which can do Fourier transform of logarithmic presentation directly. However, after the derivation of equation 5.3, the Fourier transform of logarithmic representation can be found from the original image pixel array $f(x)$ directly.

5.3 Rotation Detection

The dilation detection can be extended to two dimensional in order to detect rotation as well. A function $f(x, y)$ is transformed to its logarithmic-polar coordinate representation $\tilde{f}(r, t)$ by substituting $x = e^r cos\ t$ and $y = e^r sin\ t$. This $\tilde{f}$ function is undergone a Fourier transform and phase angle comparison. The Fourier transform under logarithmic-polar coordinate is called Fourier Mellin transform [25].

$$\tilde{F}(u, v) = \int_{-\infty}^{\infty} \int_{0}^{2\pi} \tilde{f}(r, t) e^{-j(ur+vt)} dt dr \tag{5.5}$$

With variable changing, again, $\tilde{F}(u, v)$ can be calculated from $f(x, y)$ directly. The relationship between $drdt$ and $dxdy$ is

$$\begin{aligned} rdrdt &= dxdy \\ \Rightarrow drdt &= \frac{1}{r} dxdy \end{aligned}$$

Then by substituting

$$\begin{aligned} r &= log\sqrt{(x^2+y^2)} \\ &= \frac{1}{2}log(x^2+y^2) \\ t &= tan^{-1}\left(\frac{y}{x}\right) \end{aligned}$$

We have,

$$\begin{aligned} &\tilde{F}(u,v) \\ &= \int_{-\infty}^{\infty}\int_{0}^{2\pi} \tilde{f}(r,t)e^{-j(ur+vt)}dtdr \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j(ur+vt)}\frac{1}{r}dxdy \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j(u\frac{1}{2}log(x^2+y^2)+vtan^{-1}\frac{y}{x})}\frac{1}{\frac{1}{2}log(x^2+y^2)}dxdy \end{aligned} \quad (5.6)$$

Similar to previous section, in discrete calculation, summation is used instead of integration. As a result, equation 5.6 is changed to

$$\tilde{F}(u,v) = \sum_x\sum_y f(x,y)e^{-j(u\frac{1}{2}log(x^2+y^2)+vtan^{-1}\frac{y}{x})}\frac{1}{\frac{1}{2}log(x^2+y^2)}dxdy \quad (5.7)$$

Now, suppose the rotated version of $f(x,y)$ is $f_r(x,y)$, ie, $f_r(x,y) = \mathcal{R}_\theta\big(f(x,y)\big)$, where θ is the rotation angle. Expressing $f_r(x,y)$ in logarithmic-polar coordinate,

$$\tilde{f}_r(r,t) = \tilde{f}(r,t+\theta)$$

θ can be calculated by performing Fourier Mellin transform on both $f(x,y)$ and $f_r(x,y)$, and then comparing their phase angles. Suppose $\tilde{F}(u,v)$ and $\tilde{F}_r(u,v)$ are the Fourier Mellin transform of $f(x,y)$ and $f_r(x,y)$ respectively.

Then

$$\tilde{F}_r(u,v) = e^{j2\pi\theta v}\tilde{F}(u,v)$$

$$\Rightarrow \frac{\tilde{F}_r(u,v)}{\tilde{F}(u,v)} = e^{j2\pi\theta v}$$

Let

$$G_r(u,v) = \angle\frac{\tilde{F}_r(u,v)}{\tilde{F}(u,v)} \tag{5.8}$$

The function $G_r(u,v)$ is a two-dimensional function. By fixing the variable u in the function, a curve with varying v can be plotted. In the plotting, it is a straight line with slope $2\pi\theta$. Then the rotation angle can be found. Table 5.3 shows the angle detection results.

Rotation angle (in radian)	Detected rotation angle (in radian)	Percent of error
0.001745	0.001789	2.5%
0.003491	0.003236	7.3%
0.005236	0.005147	1.7%
0.006981	0.006652	4.7%
0.008727	0.008512	2.5%
0.012217	0.012299	0.7%
0.017453	0.017177	1.6%
0.034907	0.033156	5.0%
0.052360	0.049610	5.3%
0.069813	0.065651	6.0%
0.087266	0.081179	7.0%
0.174532	0.169993	2.6%

Table 5.3: Rotation angle detection using logarithmic-polar coordinate and phase angle comparison.

The back rotation result is similar to the two cases above. The watermark is recovered and the correlation values after back rotation are ranging from 0.3286 to 0.3499.

Similar to the dilation detection, the most difficult part in the implementation of rotation detection is on finding Fourier Mellin transform. There is no function in MatLab which can find out Fourier Mellin transform directly. But the calculation can be found out from equation5.7 by plugging in the image pixel array $f(x, y)$ in it. However, there are two issues have to be considered in the implementation, they are the center of rotation and the contribution of the pixels around the rotation to the result of Fourier Mellin transform.

In rotation, there is a center of rotation. The definition of the x-y coordinate in equation 5.7 should be consistent with center of rotation. That is, the center of rotation should be defined as the origin in the x-y coordinate. In this way, the rotation center can be arbitrary and need not be fixed. But the rotation detection must be done on a known center rotation. In the real situations, it is rare for the watermark enemies to rotate an image in a way that the rotation center is known by public. For the rotation which is center-unknown, it is equivalent to a rotation followed by translation. It is a mixed geometric distortion detection and will be discussed in next chapter.

From equation 5.7, it can be found that the contribution of each pixel in $f(x, y)$ on the result, $\tilde{F}(u, v)$, is inversely proportional to the logarithm of its distance from the center of rotation, ie, $log\ r$ or $log\sqrt{(x^2 + y^2)}$. The pixels near the center of rotation contribute more and far from the center of rotation contribute less. In this way, the round-up error in calculating the logarithmic-polar coordinates of the pixels around center of rotation is magnified by a factor

of $\frac{1}{r}$ in equation 5.7 and results in large distortion of output. We found that this distortion added noise on the straight-line plot of $G_r(u, v)$ (equation 5.8) and make it impossible to find out its slope. In order to reduce this distortion, a hole is dug in the image. The pixels around the rotation center are blacked and the errors in logarithmic-polar transforming of the pixels near the center are vanished. In other words, only the pixels with less contribution to $\tilde{F}(u, v)$ or $\tilde{F}_r(u, v)$ are used in calculating Fourier Mellin transform and the result is affected less by the round-up errors in logarithmic-polar transformation of the pixels. For the size of this blacking area, it can be found by try-and-error. For an image has size 512 by 512, an area of size 100 by 100 around the center of rotation is blacked.

Chapter 6

Further work

From the previous chapter, it is found that the technique of phase angle comparison cannot cope with large scale distortion, but work very well in small scale distortion detection. As a result, this technique of phase angle comparison can be used in fine tuning the detection. Moreover, the derivations in the previous chapter were capable in dealing with separated distortion. When there is mixed distortion, the detection technique used in single distortion is not enough.

6.1 Large scale distortion detection

For the large scale distortions such as large angle rotation (rotation angle larger than 10 degree), phase angle comparison cannot help to find out the distortion magnitude. It is because the border of the distorted image is changed in this large scale distortion. Such change of border can affect the Fourier transform (or Fourier Mellin transform) of the image and affect the results in phase angle comparison.

As a result, for large scale distortion detection, other techniques have to be used. An intuitive technique is detecting by comparing the content of image. It is common for an image contains objects that have edge- or corner-like patterns. By comparing the displacement of these edges or corners, the distortion magnitude can be found approximately. The idea is similar to interactive method in section 4.2.2. When large scale distortion has to be detected and corrected, this content comparing technique can co-operate with phase angle comparison. Using content comparing technique, the rotation angle can be detected and corrected with an error within 1-2 degrees. Then it is further fine tuned by phase angle comparison with the error dropping to 0.1-0.2 degree, which is good enough to restore the correlation value in watermark detection above the threshold and recover the watermark. (figure 3.7)

6.2 Mixed geometric distortions

When there is mixed geometric distortion, the detection becomes much more complicated. For instance, translation and rotation mixed together becomes an unknown center rotation. In this way, the center of rotation has to be found out first. Since the image is also rotated, the phase angle comparison technique cannot be used to detect the translation magnitude. Similarly the combination of dilation and translation creates the same problem. However, combination of dilation and rotation can still be dealt with phase angle comparison technique and Fourier Mellin transform. In the rotation angle detection, suppose $f_r(x, y)$ is a rotated and dilated version of $f(x, y)$. Then the function $G_r(u, v)$ in equation 5.8 is modified to

$$G_r(u,v) = \angle \frac{\tilde{F}_r(u,v)}{\tilde{F}(u,v)} = \angle(e^{j2\pi\theta v} e^{j2\pi u ln\rho})$$

Therefore, if $G_r(u,v)$ is plotted with fixed v and varying u, the slope of the plot is $2\pi ln\rho$. In this way, the dilation factor can also be found.

Chapter 7

Conclusion

Digital watermark is a perceptual and statistical undetectable secondary signal hiding in the original signal. It contains the information of the owner of a digital data and it is usually used as a proof of the ownership. Although digital watermark cannot prevent a digital data from being illegally duplicated, it can be evidence when piracy is caught.

There are two categories of watermark embedding and detecting techniques: spatial domain based and frequency domain based watermark. Frequency domain based watermark has higher capacity and robustness. As a result, a frequency domain based watermarking system was implemented for doing tests in this thesis.

Robustness is an important property of digital watermark. Even the unauthorized users cannot access and detect the watermark, they can destroy the watermark by adding distortions on the digital data. A well-designed watermark should be robust to common attacks such as common signal processing attacks, collision and forgery attacks and geometric distortion attacks.

The main focus of this thesis is on defending the geometric distortion attacks on image-watermark. The geometric distortions include translation, dilation and rotation. There are existing techniques that combating these geometric distortions, such as ring-shaped watermark, watermarking with phase Taylor invariance and geometric distortion invariant watermark. They all belong to special designed watermark that resistant to geometric distortions. There is another technique of combating these geometric distortions, which is distortion detection. The distortion level is detected and corrected by doing inverse translation, dilation and rotation. Compare with the special designed watermarks, its advantage is the watermark embedding and retrieving algorithms need not be changed. The algorithms that proved to be robust can still be used, which with additional robustness on geometric distortions.

Distortion detection can be done by phase angle comparison, which has average error less than 3%. Phase angle comparison can be used directly to detect translation. With logarithmic-polar coordinate and Fourier Mellin transform, dilation and rotation can be represented as translation. Then dilation and rotation can also be detected by phase comparison. After the distortion level is detected, the distortion is corrected by inverse transform. From the experimental results, the watermark is recovered successfully after the distortion is corrected.

The distortion detection by phase angle comparison is much more complicated when the distortions are mixed. Moreover, this phase angle comparison technique cannot deal with large distortion magnitude. As a result, detecting large distortion magnitude and mixed distortion are two major further work of combating geometric distortion by distortion detection.

Bibliography

[1] Simon Singh. *The Code Book, the Evolution of Secrecy from Mary Queen of Scots to Quantum Cryptography.* 1st edition, New York: Doubleday, 1999, pp. 3-6

[2] A. Z. Tirkel, G. A. Rankin, R. M. van Schyndel, W. J. Ho, N. R. A. Mee, C. F. Osborne. *"Electronic Water Mark". Digital Image Computing, Technology and Applications – DICTA 93.* Maquarie University, 1993, pp. 666-673.

[3] H. Inoue, A. Miyazaki, A. Yamamoto, T. Katsura. *A gidital Watermark Based on the Wavelet Transform and its Robustness on Image Compression.* Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on Volume: 2, 1998, pp. 391-395 vol.2

[4] J. Fridrich. *Image watermarking for tamper detection.* Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on Volume: 2, 1998, pp. 404-408 vol.2

[5] R. Dugad, K. Ratakonda, N. Ahuja. *A new wavelet-based scheme for watermarking images.* Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on Volume: 2, 1998, pp. 419-423 vol.2

[6] A. Piva, M. Barni, F. Bartolini, V. Cappellini. *DCT-based watermark recovering without resorting to the uncorrupted original image.* Image Processing, 1997. Proceedings., International Conference on Volume: 1, 1997, pp. 520-523 vol.1

[7] W. Tang, Y. Aoki. *A DCT-based coding of images in watermarking.* Information, Communications and Signal Processing, 1997. ICICS., Proceedings of 1997 International Conference on Volume: 1, 1997, pp. 510-512 vol.1

[8] J. J. K. O. Ruanaidh, W. J. Dowling, F. M. Boland. *Watermarking digital images for copyright protection.* Vision, Image and Signal Processing, IEE Proceedings - Volume: 143 4, August 1996, pp. 250 -256

[9] A. G. Bors and I. Pitas. *Image Watermarking Using DCT Domain Constraints.* 1996 IEEE International Conference on Image Processing (ICIP'96), Lausanne , Switzerland, vol. III, pp. 231-234, 16-19 September 1996

[10] P. Bassia, I. Pitas. *Robust Audio Watermarking in the time-domain.* Proc. of EUSIPCO'98, September 8-11, Rhodes, Greece, 1998

[11] J. J. K. O. Ruanaidh, S. Pereira. *A secure robust digital image watermark.* Electronic Imaging: Processing, Printing and Publishing in Colour, SPIE Proceedings, Zrich, Switzerland, May 1998

[12] I. J. Cox, J. Kilian, T. Leighton, T. Shamoon. *Secure Spread Spectrum Watermarking for Multimedia.* IEEE Trans. Image Processing, Volume 6, No. 12, pp. 1673-1687, 1997

[13] J. J. K. O. Ruanaidh, W. J. Dowling, F. M. Boland. *Watermarking digital images for copyright protection.* IEE Proceedings on Vision, Signal and Image Processing, 143, 4, pp. 250-256, August 1996

[14] R. G. van Schyndel, A. Z. Tirkel, C. F. Osborne. *A digital watermark.* Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference Volume: 2, 1994, pp. 86-90 vol.2

[15] R. B. Wolfgang, E. J. Delp. *A watermark for digital images.* Proc. Int. Conf. On Image Processing 1996 volume 3, pp. 219-222

[16] I. Pitas. *A Method for Signature Casting on Digital Images.* 1996 IEEE International Conference on Image Processing (ICIP'96), Lausanne, Switzerland, vol. III, pp. 215-218, 16-19 September 1996

[17] N. Nikolaidis, I. Pitas. *Copyright protection of images using robust digital signatures.* IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-96), vol. 4, pp. 2168-2171, May 1996

[18] D. Kundur, D. Hatzinakos. *A robust digital image watermarking method using wavelet-based fusion.* Image Processing, 1997. Proceedings., International Conference on Volume: 1, 1997, pp. 544-547 vol.1

[19] H. Kii, J. Onishi, S. Ozawa. *The digital watermarking method by using both patchwork and DCT.* Multimedia Computing and Systems, 1999. IEEE International Conference on Volume: 1, 1999, pp. 895 -899 vol.1

[20] A. Z. Tirkel, C. F. Osborne, R. G. van Schyndel. *Image watermarking-a spread spectrum application.* Spread Spectrum Techniques and Applications

Proceedings, 1996., IEEE 4th International Symposium on Volume: 2, 1996, pp. 785 -789 vol.2

[21] S. D. Servetto, C. I. Podilchuk, K. Ramchandran. *Capacity issues in digital image watermarking.* Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on Volume: 1, 1998, pp. 445-449 vol.1

[22] W. Zhu, Z. Xiong, Y. Q. Zhang. *Multiresolution watermarking for images and video: a unified approach.* Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on Volume: 1, 1998, pp. 465-468 vol.1

[23] J. J. K. O. Ruanaidh, T. Pun. *Rotation, scale and translation invariant spread spectrum digital image watermarking.* Signal Processing, 66, 3, pp. 303-317, May 1998

[24] V. Solachidis, I. Pitas. *Circularly symmetric watermark embedding in 2-D DFT domain.* Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on Volume: 6, 1999, pp. 3469-3472 vol.6

[25] R. D. Brandt, F. Lib. *Representations that uniquely characterize images modulo translation, rotation and scaling.* Pattern Reconition Letters, 17:1001-1015, August 1996